Journal of King Saud University – Computer and Information Sciences (2016) **28**, 27–36

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

Privacy preserving cloud computation using Domingo-Ferrer scheme

Abdulatif Alabdulatif [a,b,*], **Mohammed Kaosar** [c]

[a] *School of Computer Science, RMIT University, Melbourne, Australia*
[b] *The School of Computer Science, Qassim University, Saudi Arabia*
[c] *Computer Science Department, Effat University, Saudi Arabia*

Received 21 October 2014; revised 20 October 2015; accepted 22 October 2015
Available online 6 November 2015

KEYWORDS

Homomorphic encryption;
Arithmetic operations;
Maximum/minimum function;
Cloud computing;
Cloud-based applications

Abstract Homomorphic encryption system (HES) schemes are anticipated to play a significant role in cloud-based applications. Moving to cloud-based storage and analytic services securely are two of the most important advantages of HES. Several HES schemes have been recently proposed. However, the majority of them either have limited capabilities or are impractical in real-world applications. Various HES schemes provide the ability to perform computations for statistical analysis (e.g. average, mean and variance) on encrypted data. Domingo-Ferrer is one scheme that has privacy homomorphism properties to perform the basic mathematical operations (addition, subtraction and multiplication) in a convenient and secure way. However, it works only in the positive numbers' range which is considered as a limitation because several applications require both positive and negative ranges in which to work, especially those that have to implement analytical services in cloud computing. In this paper, we extend Domingo-Ferrer's scheme to be able to perform arithmetic operations for both positive and negative numbers. We also propose using a lightweight data aggregation function to compute both maximum and minimum values of the aggregated data that works for both positive and negative numbers.
© 2015 The Authors. Production and hosting by Elsevier B.V. on behalf of King Saud University. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

The essential idea behind the efforts to involve homomorphic encryption system (HES) techniques in practical applications is to employ the advantages of cloud computing services and resources, as these techniques provide a convenient and secure environment for uploading private information to a cloud. HES has existed since the inception of public key cryptography. Examples of HES include Rivest et al. (1978), ElGamal (1985), Benaloh (1994) and Paillier (1999), to name a few. However, these are somewhat homomorphic encryption systems (SHES), meaning they only support either addition or

* Corresponding author at: School of Computer Science, RMIT University, Melbourne, Australia.
E-mail addresses: abdulatif.alabdulatif@rmit.edu.au (A. Alabdulatif), mkaosar@effatuniversity.edu.sa (M. Kaosar).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2015.10.001
1319-1578 © 2015 The Authors. Production and hosting by Elsevier B.V. on behalf of King Saud University.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

multiplication, not both. A number of these schemes fail to be appropriate for practical applications because they are often inefficient and are unable to perform many required arithmetic operations to build useful applications in cloud environments.

A fully homomorphic encryption system (FHES) would support several operations simultaneously. This was a problem until the breakthrough result of Gentry (2009), which is based on the properties of ideal lattices (Bayer-Fluckiger, 2002). This scheme is still quite impractical for real-life applications because of its limitations in arithmetic operations, time consumption and the amount of resources that are required for computations. Shortly after, a FHES was introduced that used only elementary arithmetic operations (Van Dijk et al., 2010). While this modified scheme reduced the complexity of a fully homomorphic encryption process by allowing it to be described in simple terms, it is still complex enough to be useful in any real-life applications. The aforementioned FHES schemes and other similar schemes, such as Gentry (2009, 2010), have a common drawback, which includes computational overhead in terms of efficiency and execution time. This led to the conclusion that these schemes are ineffective for real-life applications, especially in a cloud-based environment.

The Domingo-Ferrer encryption scheme (Domingo-Ferrer, 2002) is considered a lightweight scheme that has the ability to perform various arithmetic computations in a secure manner based on homomorphic properties, and it can be a possible candidate in various practical cloud-based applications. We believe that the Domingo-Ferrer's additive and multiplicative privacy homomorphism scheme (Domingo-Ferrer, 2002) is one of the most applicable HESs that can perform main basic arithmetic operations, which include addition, subtraction and multiplication. This is done in appropriate and secure ways such as through statistical analysis services in wire and wireless sensor networks (WSN), where aggregated data are analysed using aggregation functions. Indeed, it has a convenient encryption/decryption mechanism, which helps with use in various cloud-based applications (see Fig. 1).

We are working to adapt Domingo-Ferrer's scheme (Domingo-Ferrer, 2002) to be able to operate within practical applications in cloud-based environments by improving their capabilities to encompass a wider range of arithmetic operations, which leads to increased opportunities to move many of the existing applications to the cloud. In this paper, we highlight and address the following issues:

- An ability to involve a negative number's range in Domingo-Ferrer's scheme (Domingo-Ferrer, 2002). We extend Domingo-Ferrer's scheme to be able to perform arithmetic operations in both positive and negative numbers. We reorganise encryption/decryption parameters in a way that helps to carry out numbers' signs. This contribution allows the deployment of many applications that require both positive and negative ranges in cloud-based environments in a secure manner. In real-life situations, we most often think of negative numbers when we speak of real-world applications, such as military navigation systems, human health monitoring systems and many others. We have to consider how they can be manipulated in a secure and efficient way, especially those applications that involve sensitive and private data. A negative number's range on encrypted data with homomorphism properties can contribute to securing many aggregation functions that use a negative number range as an essential part of their computations.
- According to the previous contribution, we introduce an aggregation function to compute maximum and minimum values among aggregated data based on both positive and negative number ranges. This aggregation function is based on Domingo-Ferrer's additive and multiplicative privacy homomorphism scheme. We improve an idea that is shown in Ertaul and Kedlaya (2007). This is based on Domingo-Ferrer's scheme of finding maximum and minimum values among aggregated values through an ability to combine the two processes to find maximum and minimum in a single process rather than complete them separately. This is a result of taking advantage of both positive and negative number ranges instead of working in a positive numbers range only. This aggregation function is compatible to be applied to the cloud because of its ability to find maximum and minimum values among a set of values in their encrypted form without the need to reveal any information during the implementation of this function.

We revisit previous work and background concepts in Section 2. The proposed extended scheme is described in Section 3. In Section 4, we illustrate arithmetic and logical operations and their specifications based on the proposed scheme. We present implementation details and performance analysis in Section 5. Finally, we conclude the paper in Section 6.

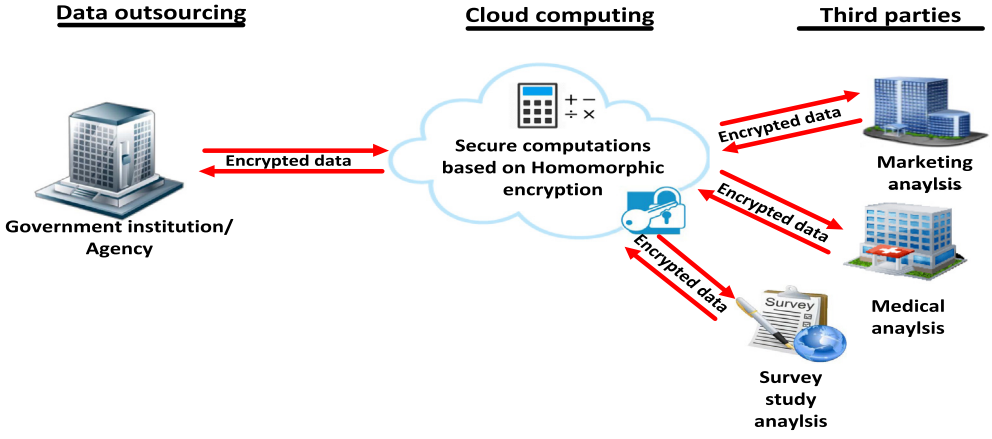

Figure 1 Numerous applications can delegate their data processing to the cloud based on HES.

Download English Version:

https://daneshyari.com/en/article/483935

Download Persian Version:

https://daneshyari.com/article/483935

Daneshyari.com